Fehlinformationen durch Kontext erkennen

Caspar Pagel Vincent Elster

Caspar Pagel (17) & Vincent Elster(17)
Erarbeitungsort: Privat(zu Hause)
Thema des Projekts: Künstliche Intelligenz, Fehlinformationen
Fachgebiet: Mathematik/Informatik
Wettbewerbssparte: Jugend forscht
Bundesland: Hamburg
Wettbewerbsjahr: 2023

1 Kurzfassung

Um falsche Behauptungen mit Hilfe von maschinellen Lernen zu erkennen ist Stance Detection (SD) eine geeignete Technik, bei der der Standpunkt zwischen einer Behauptung und Nachrichten-Kontext erkannt wird. Daher untersuchen wir verschiedene Methoden zur Integration von Kontext in SD und wie diese sich auf die Leistung eines von uns entwickelten Systems zum Erkennen von Fehlinformationen auswirken.

2 Inhalt

1 Kurzfassung **2**

2 Inhalt **2**

3 Einleitung **3**

4 Vorgehensweise, Materialien und Methode **4**
4.1 Verstehen der Mechanismen von Fehlinformationen: Herausforderungen und Lösungsansätze 4
4.2 Verwendung von Daten mit Kontext für Fehlinformations-Erkennung durch Stance Detection 4
4.3 Tf-Idf, BOW und Cosine-Similarity 5
4.4 Neue Erkenntnisse in der Desinformation Erkennung durch Austausch mit Experten und Anwendung von Self-Attention Transformer 6
4.5 Herausforderungen und Lösungsansätze der Desinformationserkennung 7

5 Ergebnisse **7**
5.1 Experimente und verwendete Metriken 7
5.2 BOW-TF, GloVe und Sentence-Transformers im Vergleich 8

6 Ergebnisdiskussion **8**
6.1 Einordnung in die Ergebnisse der Fake News Challenge 9
6.2 Zusammenhang von Verlust und F1-Score 9
6.3 Auswirkungen auf zukünftige Arbeiten 10

7 Zusammenfassung **10**

8 Quellen- und Literaturverzeichnis **11**

9 Unterstützungsleistungen **11**

3 Einleitung

Fehlinformation, oder falsche Informationen, können in vielen Bereichen schädliche Auswirkungen haben. Besonders solche, die bewusst politische, wirtschaftliche oder ideologische Interessen vertreten (Desinformationen), stellen ein zunehmendes Problem in den sozialen Medien dar [9].

Fehlinformation kann auch in politischen Ereignissen schädlich sein, denn falsche Behauptungen über Wahlergebnisse oder politische Entwicklungen können dazu führen, dass Menschen ihr Vertrauen in die demokratischen Institutionen verlieren und sogar zu gewalttätigen Auseinandersetzungen beitragen [10]. Ein bekanntes Beispiel dafür ist der Sturm auf das US-Kapitol am 6. Januar 2021, welche durch falsche Behauptungen über den Ausgang der Präsidentschaftswahlen angefeuert wurde [4]. Laut einer Studie des Oxford University's Reuters Institute for the Study of Journalism [5], trugen sowohl traditionelle Medien als auch soziale Medien dazu bei, diese Falschinformationen zu verbreiten. Desinformationen wurden in der Vergangenheit dafür genutzt, unter politischen Absichten, soziale Spannung auszunutzen, indem organisiert Minderheiten über soziale Medien attackiert und belästigt wurden[1].
Ein weiteres Beispiel sind falsche Informationen über COVID-19 und Ansteckungsrisiken oder die Wirksamkeit von invaliden Behandlungen. Dies hat dazu geführt, dass manche Menschen sich nicht an Empfehlungen zur Eindämmung der Pandemie halten und sich selbst und andere gefährden [7]. Es ist wichtig, dass wir uns bemühen, zuverlässige und validierte Informationen zu konsumieren und zu teilen, um die Ausbreitung von Fehlinformation zu verhindern und um Entscheidungen treffen zu können.

Während unserer Recherche zu möglichen Lösungsansätzen durch Künstliche Intelligenz (KI) stellte sich Stance Detection als eine geeignete Methode heraus [8]. Hier wird maschinelles Lernen genutzt, um den Standpunkt von faktischem Kontext, wie Nachrichtenartikeln, gegenüber einer Behauptung zu erkennen. Es wird unterschieden zwischen:

- Zustimmen (agree)
- Nicht zustimmen (disagree)
- Diskutieren (discuss)
- Kein Zusammenhang (unrelated)

Agree bedeutet, dass die Texte sich inhaltlich zustimmen, wohingegen disagree eine inhaltliche Ablehnung bedeutet. Bei discuss besteht zwar eine Korrelation zwischen den Texten, es kann jedoch kein klarer Standpunkt erkannt werden. Bei unrelated handeln die Texte von vollkommen unterschiedlichen Themen, wodurch ebenfalls kein klarer Standpunkt erkannt werden kann.
Bei unserer eigenen Implementationen ist uns allerdings aufgefallen, dass es schwierig ist diese beiden Texte einem numerischen Format darzustellen und gleichzeitig die für die Erkennung von Fehlinformationen relevanten Informationen beizubehalten.

In dieser Arbeit wollen wir daher den Fragen nachgehen, welche Methoden es zum Einbetten von Kontext in SD gibt und wie diese sich auf unsere Implementation eines neuronalen Netzes zum Erkennen von Fehlinformationen auswirken. Hierbei vermuten wir, dass neuere und komplexere Verfahren semantische Korrelationen zwischen der Behauptung und dem Kontext besser darstellen, was sich wiederum positiv in der Evaluation unseres Modells widerspiegeln sollte.

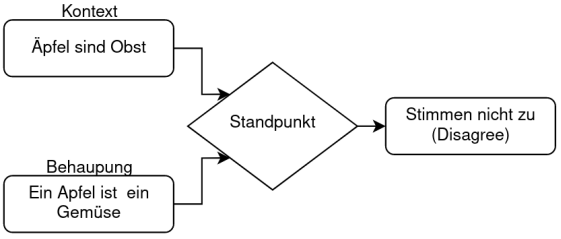


Abbildung 1: Beispiel für Stance Detection

4 Vorgehensweise, Materialien und Methode

In diesem Abschnitt werden wir unsere Vorgehensweise, die verwendeten Materialien und Methoden in unserem Projekt präsentieren. Wir werden zunächst die Schritte erläutern, die wir unternommen haben, um einen geeigneten Datensatz für unsere Analyse zu erstellen und diesen anschließend vorzubereiten. Danach werden wir die verwendeten Machine-Learning Techniken und Modelle beschreiben, um die Erkennung von Fehlinformationen durchzuführen, sowie die Methoden zur Integration von Kontext. Dieser Abschnitt gibt einen umfassenden Überblick über die Methoden und Materialien, die wir in unserem Projekt verwendet haben und bietet eine Grundlage für die Interpretation und Bewertung der Ergebnisse in den folgenden Abschnitten.

4.1 Verstehen der Mechanismen von Fehlinformationen: Herausforderungen und Lösungsansätze

In unserem Projekt haben wir uns mit der Erkennung von Fehlinformationen beschäftigt. Daher war es unsere primäre Aufgabe, ein Verständnis dafür zu entwickeln, wie solche Informationen produziert werden, wer dafür verantwortlich ist und welche Techniken und Methoden dafür verwendet werden.
Um dieses Ziel zu erreichen haben wir zunächst Literatur, speziell wissenschaftliche Artikel, zu dem Thema gelesen.
So stellte sich beispielsweise heraus, dass die Verbreitung von Desinformation oft durch emotionale Resonanz und die Verbindung zu tief verwurzelten Überzeugungen angetrieben wird. [3] Daher ist es wichtig, dass wir uns mit den emotionalen und persönlichen Aspekten von Desinformation auseinandersetzen und uns auf die Entwicklung von Methoden konzentrieren, die in der Lage sind, sowohl emotionale als auch Fakten-basierte Aspekte von Desinformation zu erkennen und zu bekämpfen. Dies erfordert eine interdisziplinäre Herangehensweise, die sowohl technische als auch sozialwissenschaftliche Methoden einschließt, um die komplexen Mechanismen von Desinformation zu verstehen und effektiv bekämpfen zu können. Wir merkten, dass wir mit unserem Ansatz nur einen Teil der Auswirkungen von Desinformationen bekämpfen können, den Fakten-Beweis, auf welchen wir uns in dieser Arbeit konzentrieren werden.

4.2 Verwendung von Daten mit Kontext für Fehlinformations-Erkennung durch Stance Detection

Behauptung (Überschrift)	**"Robert Plant Ripped up $800M Led Zeppelin Reunion Contract"**
Ausschnitt des Kontexts (Artikel)	*"... Led Zeppelin's Robert Plant turned down £500 MILLION to reform supergroup. ..."*
Korrekte Klassifikation	Agree

Abbildung 2: Beispiel eines Datenpunktes aus dem Fake News Challenge Datensatz

Der naive Ansatz der Fehlinformations-Erkennung verwendet Daten, in denen Texte als jeweils „Wahr" oder „Falsch" klassifiziert werden sollen. Es wird also probiert, diese allein durch den Ursprungstext und anhand der Satzstruktur zu erkennen.
Bei unserer Recherche wurde uns aber klar, dass falsche Behauptungen in Realität von der Satzstruktur her meist wahren Informationen ähneln [3]. Der Wahrheitsgehalt einer Information entsteht nur durch den Kontext. Daraufhin haben wir verschiedene bestehende Ansätze analysiert [8] und sind auf die Stance Detection (SD) sowie einen passenden Datensatz aufmerksam geworden.

Der Datensatz stammt von der Fake News Challenge [6] und besteht aus Überschriften von Nachrichtenartikeln 2, welche Behauptungen darstellen, und den Texten dieser Artikel.
Wenn eine Überschrift (Behauptung) dem Inhalt des Arikels (Kontext) entspricht, also zustimmt, repräsentiert das eine durch Kontext unterstützte Behauptung, die somit als „Wahr" gilt. Da es sich um Supervised-Learning handelt, wurden die Behauptung-Kontext Paare in dem Datensatz jeweils mit „Agree", „Disagree", „Discuss" und „Unrelated" von den Autor:innen der Fake News Challenge beschriftet. Dieser Datensatz ermöglicht es uns, die Standpunkte zwischen Aussagen und bestimmten Kontext zu untersuchen

und somit eine Aussage über die Wahrheit einer Aussage treffen zu können. Es war erforderlich, den

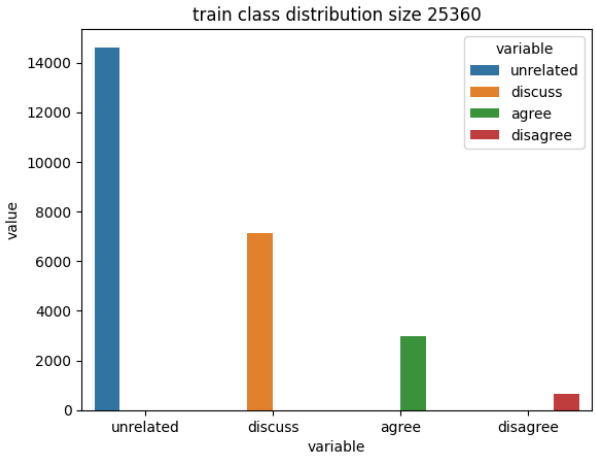


Abbildung 3: Unausgeglichene Datenmenge pro Klasse im training-Datensatz

Datensatz zu säubern und zusammenzufügen. So konnten wir sicherstellen, dass der Datensatz korrekt und vollständig war und keine irrelevanten oder fehlerhaften Daten enthielt. Um dies zu erreichen, haben wir zunächst die Daten manuell überprüft und entfernt, die nicht für die Analyse relevant waren. Anschließend haben wir die verbleibenden Daten zusammengeführt, um einen einzelnen, vollständigen Datensatz zu erstellen.
Da dieser sehr unausgeglichen war 3 haben wir die Daten per Random-Weighted-Sampling gewichtet, um sicherzustellen, dass sie repräsentativ für die untersuchte Population waren. Dieser Prozess der Datensäuberung, -zusammenführung und -gewichtung sorgt dafür, dass die Ergebnisse unserer Analyse nicht durch Unregelmäßigkeiten in den Daten verfälscht werden.

4.3 Tf-Idf, BOW und Cosine-Similarity

Anschließend haben wir uns an die Konstruktion unseres Prozesses gemacht. Da sich diese Arbeit auf die Auswirkungen der Text- und Kontextrepräsentation fokussiert, basiert unser Prozess auf der „tough-to-beat baseline“ der „Fake News Challenge“. [6]
Hier werden zuerst Behauptung und Kontext des jeweiligen Datenpunktes als Vektor eingebettet und dann zwischen diesen die Cosine-Similarity berechnet. Diese werden dann zusammengefügt und in die Eingabeschicht eines Multi-Layer-Perceptron (MLP) Modells gegeben. Nach dem Hidden-Layer nutzen wir Softmax, um die Ausgabe in Wahrscheinlichkeiten umzuwandeln. Von diesen wird die höchste als Vorhersage über den zur Eingabe passenden Standpunkt ausgewählt.
Dabei haben wir verschiedene Methoden, wie die Verwendung von Term-Frequency inverse document frequency (Tf-Idf), die Verwendung von Bag-of-Words-Modellen (BOW), und die Cosine-Similarity Metrik verwendet, um die Behauptungen mit dem entsprechenden Kontext zu vergleichen, um zu prüfen, wie die Behauptung zu dem Kontext steht. Die Methoden wurden von uns in Python mit dem PyTorch Framework geschrieben. TF-IDF und BOW sind beide Methoden zur Verarbeitung und Analyse von natürlichen Sprachtexten. Sie werden verwendet, um Texte so numerisch darzustellen, dass Wichtigkeit von gewissen Wörtern hervorgehoben wird.
Genauer gesagt handelt es sich bei BOW um eine Methode zur Extraktion von Merkmalen aus Textdokumenten. Sie besteht darin, alle Wörter im Dokument zu sammeln und sie in einem Vektor darzustellen, der die Häufigkeit jedes Wortes im Dokument enthält. Dieser Vektor wird als „Sack voller Wörter“ (Bag of Words) bezeichnet. BOW ignoriert die Grammatik und die Reihenfolge der Wörter im Dokument und konzentriert sich stattdessen auf die Häufigkeit jedes Wortes. Es ist eine einfache Methode zur Verarbeitung von Texten und hat bei Textklassifikations-Aufgaben und -analyse eine gute Leistung gezeigt.
TF-IDF (3) berechnet die Wichtigkeit eines Wortes in einem Dokument im Vergleich zu seiner Häufigkeit in anderen Dokumenten einer Sammlung. Es berechnet sich aus der Häufigkeit, mit der ein Wort in einem Dokument vorkommt (Term Frequency, TF (1)) und der Umkehrfunktion der Häufigkeit, mit der das Wort in allen Dokumenten der Sammlung vorkommt (Inverse Document Frequency, IDF (2)). Ein Wort, das häufig in einem Dokument vorkommt, aber selten in anderen Dokumenten vorkommt, wird als wichtiger betrachtet als ein Wort, das in jedem Dokument häufig vorkommt.

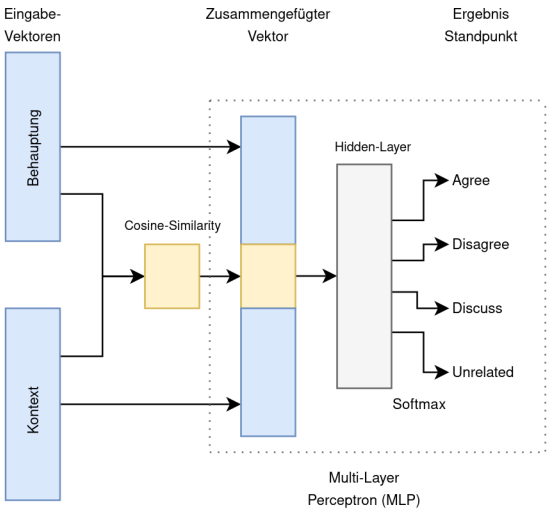


Abbildung 4: Unser MLP-Modell basierend auf der „tough-to-beat baseline"

$$tf(t,d) = \frac{t}{d} \tag{1}$$

$$idf(t,D) = log\frac{|D|}{1+|\{d \in D : t \in d\}|} \tag{2}$$

$$tfidf(t,d,D) = tf(t,d) \times idf(t,D) \tag{3}$$

t bezeichnet die Begriffe

D ist die Anzahl der Dokumente

$\{d \in D : t \in d\}$ gibt die Häufigkeit des Begriffs in Dokumenten an

Die Cosinus-Ähnlichkeit (Cosine Similarity) ist eine Methode zur Messung der Ähnlichkeit zwischen zwei Vektoren. Sie wird häufig verwendet, um die Ähnlichkeit von Texten oder Dokumenten zu bestimmen, die mit Hilfe von Methoden wie TF-IDF oder BOW in Vektoren dargestellt wurden. Die Cosinus-Ähnlichkeit berechnet sich aus dem Kosinus des Winkels zwischen den Vektoren. Sie liegt im Bereich von -1 bis 1, wobei 1 die höchstmögliche Ähnlichkeit und -1 die geringste Ähnlichkeit bedeutet. Ein Wert von 0 bedeutet, dass die Vektoren vollständig unabhängig voneinander sind. Eine der Anwendungen von Cosine similarity ist die Dokumentenähnlichkeit. Wenn zwei Dokumente ähnliche Wörter enthalten, sind die Vektoren, die diese Dokumente darstellen, tendenziell ähnlich und der Cosine-Similarity-Wert tendiert gegen 1. Wenn die Dokumente unterschiedliche Wörter enthalten, sind die Vektoren unterschiedlich und der Cosine-Similarity-Wert tendiert gegen 0.

$$cos(\theta) = \frac{v \cdot w}{\|v\|\|w\|} = \frac{\sum_{i=1}^{n} v_i w_i}{\sqrt{\sum_{i=1}^{n} v_i^2}\sqrt{\sum_{i=1}^{n} w_i^2}} \tag{4}$$

θ ist der Winkel zwischen den Vektoren **v** und **w**

v und **w** sind Vektoren

4.4 Neue Erkenntnisse in der Desinformation Erkennung durch Austausch mit Experten und Anwendung von Self-Attention Transformer

Während des Projekts haben wir auch am Bundeswettbewerb für Künstliche Intelligenz (BWKI) teilgenommen, wofür wir eine Reihe von benutzbaren Produkten, wie zum Beispiel ein Twitter-Bot und eine API, erstellt haben. Außerdem bot sich dort die Möglichkeit zum Austausch mit Experten. Durch die Erfahrungen und Anregungen, die wir dort gewonnen haben, konnten wir unsere Arbeit deutlich

verbessern und uns intensiver mit Themen wie Attention-Mechanismen auseinandersetzen. Speziell stellten sich Transformer-Modelle als ein weiteres Mittel zur Text- und Kontextrepräsentation heraus.
Sogenannte Self-Attention Transformer haben die Fähigkeit, die Beziehungen zwischen Wörtern innerhalb eines Satzes oder einer Sequenz von Sätzen besser zu verstehen. Die Verwendung von Self-Attention könnte also helfen, die Bedeutung von Wörtern im Kontext ihrer Umgebung besser numerisch darzustellen. Um die Auswirkungen von solch einer Methode auf unser System zu erforschen, ist diese Arbeit entstanden.

4.5 Herausforderungen und Lösungsansätze der Desinformationserkennung

Zusammenfassend lässt sich sagen, dass die Erkennung von Miss- und Desinformation eine komplexe Herausforderung darstellt, die sowohl technische als auch sozialwissenschaftliche Methoden erfordert. In unserem Projekt haben wir uns mit dem Verständnis von Desinformation auseinandergesetzt und uns auf die Entwicklung von Methoden konzentriert, die faktenbasierte Aspekte erkennen können. Dabei haben wir uns auf die Verwendung von Machine-Learning-Techniken und -Modellen sowie Kontext-Methoden konzentriert, um die Ergebnisse zu verbessern. Obwohl wir feststellen mussten, dass wir mit unserem Ansatz nur einen Teil der Auswirkungen von Desinformation bekämpfen können, sehen wir unsere Arbeit als wichtigen ersten Schritt in Richtung einer effektiveren Desinformations-Erkennung und -Bekämpfung an.

5 Ergebnisse

Im Anschluss werden wir unseren Experimentieraufbau und die Ergebnisse der verschiedenen Messungen beschreiben.
Die Experimente fanden unter Verwendung der gleichen Modell-Architektur statt, wodurch die einzige Veränderung in der Einbettung des Textes und Kontextes besteht. Gleichermaßen bestehen beim Test der drei Methoden folgende Hyperparameter:

- Lernrate (lr) von 0.001
- 40 Training- und Testepochen
- Batch-Größe von 64
- Hidden-Layer Größe von 100

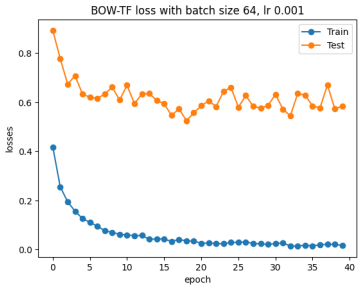
5.1 Experimente und verwendete Metriken

Um zu ermitteln, wie sich verschiedene Methoden zum Einbetten von Kontext auf unser SD-System auswirken, haben wir drei MLPs jeweils mit anderer Textrepräsentation trainiert.

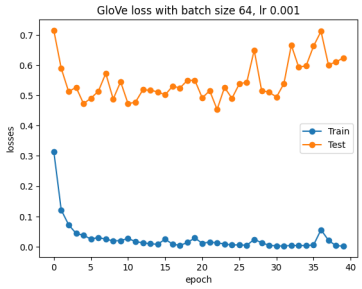
Bei den getesteten Methoden handelt es sich um BOW mit TF-IDF (BOW-TF), Sentence-Transformers (ST) und GloVe-Emeddings. Letzteres ist eine weitere beliebte Art, Text einzubetten, dessen Komplexität sich etwa zwischen BOW-TF und ST einordnen lässt.
Zur Evaluation verwenden wir Werte von Präzision und Recall. Dies sind klare und oft verwendete Metriken, durch welche sich Rückschlüsse auf den Trainingsprozess und den Umgang des Modells mit neuen Daten ziehen lassen. Da in der Regel eine Ausgeglichenheit zwischen Präzision und Recall gewünscht ist, verwenden wir zusätzlich den F1-Score, welcher folgendermaßen definiert ist:

$$F1 = 2 \cdot \frac{Präzision \cdot Recall}{Präzision + Recall} \tag{5}$$

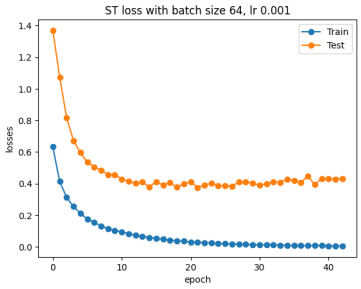
Dieser wird also größer, je dichter Präzision und Recall beieinander sind und je höher beide sind.
Dabei wurde der Training- und Testverlust pro Epoche gemessen. Trainings-Verlust zeigt an, wie groß der Fehler ist, den das Modell beim Vorhersagen des Standpunktes auf den Trainingsdaten macht. Je kleiner dieser ist, desto besser passt das Modell auf die Trainingsdaten.
Test-Verlust ist ein Maß dafür, wie gut das Modell auf neuen, ungesehenen Daten abschneidet. Es gibt an, wie groß der Fehler ist, den das Modell beim Vorhersagen der richtigen Ausgabe auf den Testdaten macht. Je kleiner dieser ist, desto besser generalisiert das Modell auf neue Daten.



(a) Epochen-Loss mit BOW-TF Einbettungen



(b) Epochen-Loss mit GloVe Einbettungen



(c) Epochen-Loss mit Sentence-Transformer Einbettungen

Abbildung 5: Epochen-Loss Vergleich

5.2 BOW-TF, GloVe und Sentence-Transformers im Vergleich

In Abbildung 4 ist die große Differenz zwischen Train- und Testverlust bei BOW-TF ein klares Indiz dafür, dass hier die Komplexität der Daten nicht gut dargestellt wird und das Modell so keine Ansatzpunkte findet, welche mit dem Standpunkt korrelieren. Dieses Phänomen entspricht unserer Erwartung, dass solche simplen Modelle sich nicht zum Erkennen von Fehlinformationen eignen.

Bei den GloVe Einbettungen zeigt sich in Anbetracht der Skala des Epochen-Verlustes eine Verbesserung, allerdings entfernen sich die Graphen ab etwa Epoche 20 bereits voneinander. Der Punkt, ab dem sie sich entfernen, wird im maschinellen Lernen als Wendepunkt bezeichnet und gilt als der Moment, ab dem das Model zu sehr an die Trainingsdaten angepasst ist und dadurch schlechter auf neue, ungesehene Daten generalisiert. Anstatt die wichtigen Muster in den Daten zu erfassen, werden Ergebnisse nur „auswendig gelernt“.
Zudem gilt hier wieder, dass die Differenz der beiden Graphen immer noch relativ groß ist, die Daten also erneut nicht optimal repräsentiert werden. Diese beiden Eigenschaften der Graphen deuten an, dass das Modell durch fehlende Komplexität nur wenig Informationen sinnvoll verarbeiten kann.
Diese im Vergleich zu BOW-TF zwar bessere, aber scheinbar immer noch suboptimale Repräsentation der Texte durch GloVe hat uns verwundert, da wir mit einem deutlichen Unterschied gerechnet haben.

Unserer Vermutung, dass neuere und komplexere Methoden zur Einbettung von Text sich positiv auf die Messungen auswirken, bestätigt sich bei den Sentence-Transformers. Es zeigt sich ein regelmäßigeres Lernverhalten und eine kleinere Differenz zwischen Train- und Testverlust. Anzumerken ist außerdem, dass sich bei allen drei Modellen ein steigender Testverlust im Bereich um 20 Epochen abzeichnet.

Die weiteren Test-Metriken haben zusätzliche Erkenntnisse geliefert. So zeigt sich in Abbildung 5, dass BOW-TF in den meisten Fällen die höchsten, also die besten Ergebnisse liefert. Interessant ist, dass diese zwar häufig denen von GloVe ähneln, letztere aber bessere Resultate in der Klasse „disagree“ vorzeigen, welche den geringsten Support hat. Support stellt dar, wie oft die jeweilige Klasse in den Testdaten vertreten ist. Dies ist verhältnismäßig zu der Anzahl im gesamten Datensatz.
Bei allen drei Modellen besteht eine sichtbare Korrelation zwischen den Messwerten und dem Support, besonders ausgeprägt allerdings bei ST. Diese erreichen bei „unrelated“, der am häufigsten vertretenen Klasse, eine Präzision von 100%, während sie bei „disagree“, der am wenigsten vertretenen Klasse, nur 23% ist.
Es ist anzumerken dass die in Abbildung 5 dargestellten Metriken jeweils nach 40 Epochen gemessen wurden und so nicht unbedingt den im Verlauf des Trainings besten Wert haben. Dennoch waren wir verwundert über den gegensätzlichen Zusammenhang zum Verlust.

6 Ergebnisdiskussion

In dieser Diskussion werden wir unsere Ergebnisse hinsichtlich der Durchführung, Interpretation und Limitationen unserer Arbeit untersuchen und diskutieren.
Zu Anfang bestanden Schwierigkeiten in der Bereinigung und Verarbeitung des Datensatzes. Die technische Umsetzung hat in der Regel reibungslos funktioniert, nur wurde der Trainingsprozess teils durch Hardware-Limitierungen und hohe Rechenkomplexität erschwert.

	precision	recall	f1	support
agree	0.87	0.93	0.90	697
disagree	0.68	0.75	0.71	152
discuss	0.97	0.95	0.96	1845
unrelated	0.99	0.99	0.99	3642

Testergebnisse mit BOW-TF-Embeddings
(lr = 0.001, 40 Epochen)

(a) BOW-TF Einbettungen

	precision	recall	f1	support
agree	0.86	0.88	0.87	702
disagree	0.75	0.81	0.78	183
discuss	0.93	0.95	0.94	1796
unrelated	0.98	0.96	0.97	3655

Testergebnisse mit GloVe-Embeddings
(lr = 0.001, 40 Epochen)

(b) GloVe Einbettungen

	precision	recall	f1	support
agree	0.50	0.79	0.61	1395
disagree	0.23	0.92	0.37	324
discuss	0.79	0.78	0.78	3649
unrelated	1.00	0.79	0.86	7304

Testergebnisse mit ST-Embeddings
(lr = 0.001, 40 Epochen)

(c) Sentence Transformers

Abbildung 6: Vergleich der Test-Metriken

6.1 Einordnung in die Ergebnisse der Fake News Challenge

In diesem Abschnitt werden die gemessenen Ergebnisse mit den Ergebnissen anderen Arbeiten an der Fake News Challenge verglichen. Die Ergebnisse anderer Arbeiten beziehen wir aus einer nachträglichen Analyse der Fake News Challenge (FNC) von der TU Darmstadt [2], in welcher die erfolgreichsten eingereichten Systeme der FNC getestet und kritisch untersucht wurden.

System	F1 (macro average)
FNC basline	.499
UCLMR	.583
stackLSTM	.609
Unser BOW-TF	**.89**
Unser GloVe	.83
Unser ST	.66

In der Tabelle findet sich der macro average F1 score der FCN baseline, das System der „tought-to-beat baseline“ des University College London, auf dessen Architektur wir auch unser System basiert haben, sowie stackLSTM, das beste der gemessene System. Verglichen mit diesen schneiden alle unsere Implementationen besser ab. Vor allem scheint dies daran zu liegen, dass bei der ursprünglichen Bewertung der FNC die unausgeglichenen Klassen 3 nicht berücksichtigt wurden, wodurch viele der Modelle die Klasse „Disagree“ nicht beachtet haben. Wir vermuten daher, dass wir dieses Problem durch das Random-Weighted-Sampling lösen konnten, was eine starke positive Auswirkung auf den F1 score unserer Systeme hat.
Auffällig ist jedoch, dass unser simpelstes System, die BOW-TF Einbettungen, insgesamt den besten F1 score aufweisen. In Anbetracht der Verlustkurven stellen wir die Vermutung auf, dass hier zwar gute Ergebnisse innerhalb des Datensatzes gelingen, wegen des festes Wörterbuches von BOW-TF jedoch eine geringe Generalisierung auf völlig neue Daten bestehen könnten.

6.2 Zusammenhang von Verlust und F1-Score

Bei allen drei Modellen zeichnet sich ab, dass bei größerer Verlust-Differenz der F1-Score pro Klasse häufig besser ist. Dieser anti-proportionale Zusammenhang ist ungewöhnlich und hat uns verwundert. Wir sind zu verschiedenen möglichen Erklärungen gekommen.
Ein Grund wäre die vergleichsweise geringe Komplexität der Architektur des MLPs der „tought-to-beat baseline“. Es kann sein, dass dieses Modell schlicht nicht in der Lage ist, ausreichende Zusammenhänge in den Daten zu erfassen. So entsteht der hohe Test-Verlust. Die häufig guten F1-Score Werte würden sich in

dem Fall daraus ergeben, dass die wenigen Zusammenhänge, die das Modell findet, auswendig gelernt werden. Diese Vermutung wird durch den Wendepunkt des Test-Verlustes bei etwa 20 Epochen gestärkt, was ein Indiz für solch ein „Overfitting“ ist.

Eine andere Möglichkeit ist, dass die Daten selber nicht genug Informationen enthalten, um den Standpunkt vorherzusagen. Beispielsweise könnten die Texte untereinander zu ähnlich sein. Die starken Unregelmäßigkeiten im Test-Verlust von BOW-TF und GloVe 4 deuten dies an. Allerdings zeigt sich bei ST eine deutliche Verbesserung in diesem Aspekt, was wiederum dafür spricht, dass die beiden anderen Einbettungen nur ungenügend Informationen erfassen konnten.
Natürlich ist eine Kombination der genannten Erklärungen ebenfalls denkbar.

Auch zu den im Vergleich zu BOW-TF deutlich schlechteren Ergebnisse von ST bei den Klasse „agree“ und „disagree“ haben wir Vermutungen aufgestellt.
Im Gegensatz zu den anderen beiden Methoden, verwendet BOW-TF nähmlich ein festes Wörterbuch basierend auf den Trainingsdaten. Alle Wörte, welche sich nicht darin befinden, werden auch im Vektor nicht repräsentiert. So passt sich das Modell sehr präzise an die spezifischen Texte beim Training an, kann aber kaum auf neue Texte generalisieren. Das erklärt den hohen Test-Verlust sowie die hohe Präzision. Gleichzeitig heißt dies, dass solch eine Einbettungsmethode sich nicht für den Anwendungsfall eignen.
Im Kontrast dazu sehen wir bei ST eine sehr starke Korrelation zwischen F1-Score und Support besteht. Wir schließen daraus, dass dieses komplexe Transformer-Modell eine größere Datenmenge benötigt, dann aber sehr gute Ergebnisse liefer.

6.3 Auswirkungen auf zukünftige Arbeiten

Die Forschungsergebnisse helfen, unsere Kontext-basierte SD Methode zur Erkennung von Fehlinformationen besser zu verstehen. Zukünftig würden wir unsere so abändern, dass wir zusätzlich zu den verschiedenen Einbettungen auch verschiedene Modelle mitberücksichtigen. Damit lässt sich verhindern, dass Eigenschaften des Modells als Eigenschaften der Einbettungen interpretiert werden.
Außerdem können die gewonnenen Erkenntnisse zu Systemen beitragen, welche die Auswirkungen von Fehlinformation in den sozialen- und traditionellen Medien minimieren könnte. Dies könnte helfen, soziale Spannungen und gefährliche Verbreitung von falschen Informationen zu verhindern. Allerdings ist es wichtig zu beachten, dass unser System noch weiterer Tests und Verbesserungen bedarf, um eine volle Funktionalität und Wirksamkeit zu gewährleisten.
Da die Verwendung von großen Sprachmodellen wie OpenAI's ChatGPT immer populärer wird, gibt es eine große Nachfrage nach besserer Integration von Kontext. Unsere Entdeckungen könnte eine Grundlage schaffen, durch die KI externen Kontext besser verstehen und bewerten kann, was zu einer Verbesserung der faktenbasierten Ergebnisse führen würde.
Zudem hat uns die Gegensätzlichkeit von Loss und F1 score angeregt, genauer auf die Qualität des verwendeten Kontexts in Relation zu der Behauptung einzugehen. So arbeiten wir Aktuell an einem System, dass qualitativen Kontext zu Behauptungen in Echtzeit aus dem Internet beschaffen kann.

7 Zusammenfassung

In dieser Arbeit haben wir Methoden zur Einbettung von Kontext für Stance Detection Systeme vorgestellt. Insbesondere wird dies verwendet, um den Standpunkt zwischen einer Behauptung und Nachrichtenkontext zu erkennen. Wir erforschen, wie sie sich die Methoden Bag of Words mit TF-IDF (BOW-TF), Sentence-Transformers und GloVe-Embeddings auf die Leistung unseres Systems zur Erkennung von Fehlinformationen auswirken.
Die Experimente haben gezeigt, dass die Verwendung von Sentence-Transformers zu dem besten Test- und Trainings-Verlust führt, GloVe und BOW-TF jedoch in anderen Metriken teils besser abschneiden. Gleichzeitig liefern alle drei Implementierungen bessere Ergebnisse als vorherige Arbeiten mit diesem Datensatz, da wir die Unausgeglichenheit des Datensatzes berücksichtigen. Dafür bieten wir mögliche Erklärungen und Kritik, anhand dessen wir Ziele für zukünftige Verbesserungen herausgearbeitet haben. Abschließend lässt sich sagen, dass die Art der Einbettungen von Kontext eine signifikanten Einfluss auf die Ergebnisse bei der Erkennung von Fehlinformationen hat und weitere Untersuchungen in diesem Bereich von großem Nutzen sein könnten, um die Erkennung dieser weiter zu verbessern.

8 Quellen- und Literaturverzeichnis

Literatur

[1] Robert Booth u. a. "Russia used hundreds of fake accounts to tweet about Brexit, data shows". In: *The Guardian* (Nov. 2017). Hrsg. von AlexEditor Hern. URL: https://www.theguardian.com/world/2017/nov/14/how-400-russia-run-fake-accounts-posted-bogus-brexit-tweets.

[2] Andreas Hanselowski u. a. "A Retrospective Analysis of the Fake News Challenge Stance-Detection Task". In: *Proceedings of the 27th International Conference on Computational Linguistics*. Santa Fe, New Mexico, USA: Association for Computational Linguistics, Aug. 2018, S. 1859–1874. URL: https://aclanthology.org/C18-1158.

[3] Kung-Hsiang Huang u. a. *Faking Fake News for Real Fake News Detection: Propaganda-loaded Training Data Generation*. 2022. DOI: 10.48550/ARXIV.2203.05386. URL: https://arxiv.org/abs/2203.05386.

[4] Rafael A Jacobs. "Elite Impact on the Capitol Hill Riot: The Straw that Stirs the Drink". Diss. 2022.

[5] Nic Newman u. a. *Reuters Institute digital news report 2021*. Techn. Ber. 2021.

[6] Benjamin Riedel u. a. *A simple but tough-to-beat baseline for the Fake News Challenge stance detection task*. 2017. DOI: 10.48550/ARXIV.1707.03264. URL: https://arxiv.org/abs/1707.03264.

[7] Hans Rosenberg, Shahbaz Syed und Salim Rezaie. *The Twitter pandemic: The critical role of Twitter in the dissemination of medical information and misinformation during the COVID-19 pandemic*. 2020. DOI: 10.1017/cem.2020.361.

[8] "Stance Detection on Social Media: State of the Art and Trends". In: (). arXiv: 2006.03644. URL: https://arxiv.org/abs/2006.03644.

[9] *The danger of fake news in inflaming or suppressing social conflict*. Aug. 2018. URL: https://www.cits.ucsb.edu/fake-news/danger-social (besucht am 22. 01. 2023).

[10] Samuel C Woolley. *Bots and computational propaganda: Automation for communication and control*. 2020, S. 89–110.

Abbildungsverzeichnis

1 Beispiel für Stance Detection 3
2 Beispiel eines Datenpunktes aus dem Fake News Challenge Datensatz 4
3 Unausgeglichene Datenmenge pro Klasse im training-Datensatz 5
4 Unser MLP-Modell basierend auf der „tough-to-beat baseline" 6
5 Epochen-Loss Vergleich 8
6 Vergleich der Test-Metriken 9

9 Unterstützungsleistungen

Wir bedanken uns bei den Juroren des BWKI 2022 für die guten Denkanstöße.